NORTHWEST SUBURBAN APPLE USERS GROUP

THE HARVEST

FEATURE ARTICLE

VOL. 2, NO. 4 DECEMBER 1980

APPLESOFT VARIABLE GOSUB

Kurt G. Schumacher
111 S. Highland
Arlington Heights, IL 60005

I recently wrote a program called "MR APPLE HEAD", which allows the person running the program to control the facial expressions of a lo-res face. (Look for this program in your local NSAUG library.)

In writing this program, I was faced with the task of obtaining a command from the user, verifying it against a list of valid commands, and going to an appropriate subroutine to process the command.

Initially, this was no problem. But as the program grew, and more and more commands were added (mostly at the instigation of my five year old son), the task of maintaining the branching logic increased greatly. The listing for Program 1 shows a simplified version of the problem. A list of valid commands is read in from DATA statements in Line 30. Lines 40-60 prompt the user for a command, verify it by looping through the command array, and issue an error message if no match is found.

Line 70 issues an ON GOSUB command to branch to the appropriate subroutine for each command. Here is where my problem arose. As the number of commands allowed by my program grew to over twenty, it began to take longer to update the ON GOSUB statement for a new command than it did to code the new subroutine! The program logic also became more difficult to follow. In order to locate the line number of the subroutine for a command, I had to count through the data statements to get the number of the command, then count through the line numbers in the GOSUB statement to find the correct subroutine. Clearly, another method was required.

I thought of the variable GOSUB feature of Integer BASIC. The listing for Program 2 shows the same logic implemented in an Integer BASIC program. The subroutine line number is stored in the array LN, which is used as the object of the GOSUB in line 60. I began to wonder if a similar method could be used in Applesoft.

I finally hit upon the idea of providing a model GOSUB statement in my program, and modifying it during program execution to point to the desired subroutine. The result is shown in the listing for Program 3. Line 0 branches around the model GOSUB. Line 30 reads in the valid commands, as before; but it also reads in the corresponding subroutine line numbers, as character strings. Lines 40-60 perform command input and validation. Lines 70-110 modify and execute the model GOSUB.

In order to understand what is going on in Lines 70-110, it is neccessary to examine the way Applesoft programs are stored in memory. Memory locations $67-$68 (103-104) point to the beginning of the program. For ROM Applesoft, this is $801. Figure 1 shows a Monitor listing of the start of Program 3.

CONTINUED ON PAGE 2

The first two bytes contain the low and high bytes of the address of the next line in memory; in this case, $808. The next two bytes are the low and high bytes of the line number, in hexadecimal ($0000). The remaining bytes in the line are the tokenised BASIC statements. In Line 0, $AB is the token for GOTO; $32 is the ASCII representation of '2'; and $00 marks the end of the line. At the beginning of Line 1 is the address of Line 2 ($815), followed by the line number ($0001). The next byte, $B0, is the token for GOSUB. The next five bytes of $30 are the ASCII representation of the line number '00000'. Finally, the next two bytes ($3A $B1) are the tokens for ':RETURN', followed by the $00 end-of-line indicator.

The portion of Line 1 that we wish to modify is the '00000', the line number that the GOSUB will branch to. The implementation in Program 3 is rather crude, but effective. Line 70 obtains the pointer to the start of the program from $67-68, and adds 11 to it. This is simply the number of bytes from the start of the program to the start of the line number to be modified, obtained by counting bytes in the Monitor listing.

Lines 80-100 poke each byte of the subroutine line number into the GOSUB statement in Line 1. Line 110 issues a GOSUB to the modified Line 1, and returns to the prompt message.

Notice the following restrictions in this technique:

-- the line number in Line 1 must have EXACTLY five zeros.

-- the subroutine line numbers in the data statements must also have EXACTLY five digits.

-- the technique as shown will only work if Lines 0 and 1 are coded exactly as shown in Program 1.

As a general rule, I would say that self-modifying BASIC programs should be avoided. In this case, however, the technique is justified by the improved readability and maintainability of the command/subroutine list.

```
10 NC = 3: REM  NUMBER OF COMMAND
     S
20  DIM CM$(NC)
30  FOR I = 1 TO NC: READ CM$(I):
      NEXT I
40  INPUT "COMMAND: ";C$
50  FOR I = 1 TO NC: IF C$ = CM$(
     I) THEN 70
60  NEXT I: PRINT "INVALID COMMAN
     D": GOTO 40
70  ON I GOSUB 200,210,220
80  GOTO 40
200  PRINT "LINE 200": RETURN
210  PRINT "LINE 210": RETURN
220  PRINT "LINE 220": RETURN
900  DATA A
910  DATA B
920  DATA C
1000  REM
```

PROGRAM 1

```
)LIST
   10 DIM LN(3)
   20 LN(1)=200:LN(2)=210:LN(3)=220

   30 INPUT "COMMAND: ",N
   40 FOR I=1 TO 3: IF N=I THEN 60

   50 NEXT I: PRINT "INVALID COMMAND"
      : GOTO 30
   60 GOSUB LN(I)
   70 GOTO 30
  200 PRINT "LINE 200": RETURN
  210 PRINT "LINE 210": RETURN
  220 PRINT "LINE 220": RETURN
  900 REM
```

PROGRAM 2

```
*801.820

0801- 08 08 00 00 AB 32 00
0808- 15 08 01 00 B0 30 30 30
0810- 30 30 3A B1 00 2E 08 02
0818- 00 B2 4C 48 4E 45 53 20
0820- 30
```

CONTINUED ON PAGE 3

FROM PAGE 2

]LIST

```
0  GOTO 2
1  GOSUB 00000: RETURN
2  REM LINES 0 & 1 MUST BE
3  REM ENTERED EXACTLY AS SHOWN
10 NC = 3: REM  NUMBER OF COMMAND
     S
20  DIM CM$(NC),GS$(NC)
30  FOR I = 1 TO NC: READ CM$(I),
     GS$(I). NEXT I
40  INPUT "COMMAND: ";C$
50  FOR I = 1 TO NC: IF C$ = CM$(
     I) THEN 70
60  NEXT I: PRINT "INVALID COMMAN
     D". GOTO 40
70 N =  PEEK (103) +  PEEK (104) *
     256:N = N + 11
80  FOR J = 1 TO 5:N$ =  MID$ (GS
     $(I),J,1)
90  POKE N + J, ASC (N$)
100  NEXT J
110  GOSUB 1: GOTO 40
200  PRINT "LINE 200": RETURN
210  PRINT "LINE 210": RETURN
220  PRINT "LINE 220": RETURN
900  DATA A,00200
910  DATA B,00210
920  DATA C,00220
1000  REM
```

PROGRAM 3

QUIZ GAME

by Peter Zaloga

Assume you are in a room that has only two doors (we'll label them 1 and 2). The only thing in this room with you are two computers (we'll label them A and B).

The problem you have is that one of the doors leads to instant death where as the other leads to safety.

The only way to have to determine which is the correct door is to direct a question. Only one question to either one of the computers.

The trouble here is that one computer always lies and the other always tells the truth. Of course, you don't know which one is telling the truth.

What is that one question you can ask that will allow you to select the correct door? Good luck!

The answer will be given at the December meeting.

****

BEER CAN AND BOOKIES

According to an AP wire service appearing in the Sun Times and Tribune a couple of weeks ago, "a computer at one of the governments top secret weapons laboratories was improperly used by 200 employees to store games, personal letters, jokes, and an illegal bookmaking operation."

The beer cans were an inventory of a collection. Other items stored were a loan repayment schedule, poetry, and a roster for the laboratories bowling team. They also found 247 games in the computer. Of course, Star Trek was one.

It seems to me they would solve all their problems by buying an APPLE. Does anyone have a book-making program?

Dave Alpert

MEMBERSHIP RENEWALS

Please look at the mailing label on this newsletter. If you have an '80' under your name, then you have not renewed for 81. Renewals for 81 must be in by December in order to continue receiving membership benefits. The mailing list will be expunged of delinquent names about the 20th of December.

Applications were included in a prior issue. If you have misplaced yours or did not receive one, please call the club secretary. Please note that renewals are $12.00 and do not require that a disk be given to the club. Renewals may be mailed to the club PO Box or dropped off at the next meeting.

We are shooting to equal or exceed last year's renewal rate of 97%. Please attend to this matter as soon as possible.

Thank You,
Your Officers

****

CLUB NEWS

CLUB OFFICERS

President: Don Fuller-------312-991-6868
Vice Pres: Mike Robins------312-593-2709
Secretary: Joel Kurasch-----312-677-8358
Treasurer: James Wilson-----312-358-2198
Librarian: Mike Rose--------312-359-4308
Asst. Lib: Julian Vassay
Prog. Crm: Ken Rose
NL Staff: Jim Nowak

Club Addresses:

MEMBERSHIP, etc--BOX 787
Palatine, Il. 60067

NEWSLETTER-------Dave Alpert
880 Mellody
Lake Forest, Il. 60045
312-295-6078
SOURCE#-TCA 640

BULLETIN BOARD---312-295-6926 daily
as available and after
10:30pm local time

Membership is open to all. Dues are $12.00 annually. New members are required to supply one disk or cash equivalent ($5.00) at the time of admission to the club. Disk not required of renewals. Membership applications are available from the club Secretary at the meetings or by mail.

Regular meetings are held at the Schaumberg Library, 100 Library Lane, Schaumberg, Il.

MEETING SCHEDULE

Regular Meeting Sat. December 6
Sat. January 3
Sat. February 7*
*tentative

Officers meetings are held at the President's home on the Tuesday evening following the regularly scheduled meeting starting at 7pm.

MEMBERS AIDE

The members listed below have volunteered to answer questions from club members who need a 'HOT LINE' type approach. Please try to be brief when you call as a courtesy to them.

Listed below are the names and phone numbers. The numbers after their names represent their special talent. PLEASE-no calls after 10pm.

Earl Allen	312-837-9259	1-9
Mike Robins	312-593-2709	9-0
Paul Stadfield	312-359-2378	4-9
Guy Lyle	312-438-7941	4-9
Ted Roseman		
Mary Roseman	312-338-4833	2,3
Joel Kurasch	312-677-8358	9
Leon Alexander	312-725-5309	9
Gordon Banks	312-324-6184	0
Rich Lundeen	312-420-2008	2,3

Mach. Lang=1 Int. Basic=2
Applesoft =3 Hardware =4
Arrays =5 DOS =6

All of above=9
PASCSAL=0 (not included in 9 designation)

When numbers appear befor a 9, the person is especially qualified in that area as well as in all other areas.

****
* *
* DECEMBER *
* AGENDA *
* DEC 6 *
* *
* *
* 10:00-10:30 OPENING REMARKS *
* Don Fuller *
* 10:30-11:15 INTERACTIVE VIDIO *
* Jim Keck *
* 11:15-12:15 NOMINATIONS *
* Joel Kurasch *
* 12:15-12:30 GAMES WE LIKE TO PLAY *
* Chuck Anderson *
* 12:30-12:55 ASK MR. APPLE *
* Mike Robins *
* 12:55- 1:00 PRESIDENT'S REMARKS *
* Don Fuller *
* *
* special mystery guest to appear *
* at this meeting *
* *
* GUESTS WELCOME *
* *
****

MORE CLUB NEWS

PREZ SEZ:

Our hat is off to Terry Tufts for the "Printer Technology" presentation this month. A lot of research went into the handout and is appreciated by all. Terry did the slides himself (!). The Program Line Editor demo by Dave Alpert was excellent. I'm sure you realized that if you're doing any programming at all, this aid is a definite M U S T !!!

No one asked, but I'm going to tell you anyway... The proposed constitution that went out will be voted on at the January meeting. Two reasons: First, that is required by the present constitution (time frame involved to get it to all members). Second, we want to set up the incoming officers with a document to incorporate the club and better define the club's operations! I will not (repeat -NOT-) read the proposal at the next two meetings due to the time constraints. you will be expected to know what you are voting on at the January meeting, so read it and think about it at your leasure. If it passes, it will be published in "THE HARVEST" and everyone will have a copy (also the 'new member' packet will have a copy for late joiners).

Nominations: Joel Kurasch will be accepting nominations from the floor at the December meeting. If you nominate someone, you must vouch for their willingness to serve in that office! (for those people who asked... No, I am not running for a second term. thank you anyway)

February 14th, 1980. The reservations are coming in for the social... Remember, if you live out of town and can't get to the meetings, this would be a good chance for a massive, once a year, get to know the people of 'nsaug'... See you there?

Oh yeah, the name change proposals in the suggestion box at the October meeting. The names suggested are already in use except for 'Northern Illinois Apple Users Group'. We may just step into this in detail at the Dec. or Jan. meeting... Be there to see what transpires!

More oh yeah: I appreciate your approval of the funds to subsidize the February social. THANK YOU.

Enough? You bet! See you December the 6th, same time, same place.

DON

SECRETARY'S REPORT

Meeting of November 1, 1980
Attendance: 135

The meeting was called to order by Program Chairman Ken Rose, who announced next month's agenda, reminded everyone of the program contest (for which there are still more prizes than entrants), and told of the plans for the the dinner dance to be held in February. President Don Fuller then gave a brief talk about the proposed new constitution of the club. Treasurer Jim Wilson reported on the club finances (see Treasurer's report). Librarian Mike Rose talked briefly about the library. And Dave Alpert, the Harvest editor, ask for anyone who was interested in working on the newsletter with him to contact him.

Terry Tufts then gave a comprehensive report on the various types of printers available and a brief discription of the technology involved for each group.

Ken Rose then explained the club is supporting the formation of various SIG's (Special Interest Group.) To further that goal, flags for the interest groups, games, music, business, stocks, etc., were distributed, so that the membership could gather by special interest after the meeting. The groups are to make reports to the club as to their progress.

Dave Alpert then gave a demonstration on the Program Line Editor. This very versitile program resides in upper memory and provides tremendous assistance to the program development effort.

TREASURERS REPORT

Month ending October 31

Sept. Ending Bal.	$1183.53
Membership	589.00
Newsletter	(158.85)
Disk Project	(577.00)
(400 disks purchased)	
Stationery	(24.80)
Bulletine board	(6.19)
Oct. Ending Bal.	$ 885.59

Respectfully submitted,
Jim Wilson, Treasurer
Nov. 4, 1980

UNIVERSAL INPUT FUNCTION FOR FORTRAN

John Russ, 213 Merwin Road, Raleigh, N.C., 27606

Apple Fortran is great, but like all Fortran's, it suffers from awkward I/O formatting structures. This is particularly annoying when inputting data from the keyboard: for instance, you have to know whether F or I format is being used to know whether you must or must not include a decimal point. To overcome this problem, the enclosed functions allow free-format input of integer (INPUT()) or floating point (VALIN()) numbers. Spaces are ignored, plus sign and decimal point are optional, and the number may appear anywhere in a 20-character field. Pressing RETURN with no character is interpreted as zero. Illegal characters cause the routine to ask for re-entry, rather than generating a fatal run-time error.

A word in addition about how to combine these functions with other programs (it took a while to figure this out, as the manual is less than clear). Assume that the functions are in files named ENTRY.CODE and ENTRY.TEXT. You can $INCLUDE ENTRY anywhere in your main program TEXT (preferably toward the end) to compile everything together just as though the ENTRY.TEXT were typed into your program TEXT, but this is both slow and inefficient. The alternative is $USES UINPUT IN ENTRY at the start of your program. The INPUT, VALIN, and TDV routines are then all found and cross-referenced during compilation. I do not know why the U prefix appears, but it looks like an abbreviation for UNIT, as in Pascal. However, at LINK time, the automatic compile-link-and-run sequence will bomb when it can't find UINPUT. You have to select L(ink from the command menu and specify both the *SYSTEM.LIBRARY and ENTRY as library files. Then everything seems to come out OK.

P.S. for Dr. Apple: How can I use the GOTOXY routines from Pascal to control my output to the console screen in Fortran?

My best to all.....

```
C     UNIT 'UINPUT' FOR GENERAL USE
C     (C) JOHN C. RUSS OCT 1980

      FUNCTION INPUT()
C     FREE FORMAT ENTRY FOR INTEGERS
      IMPLICIT CHARACTER (H), LOGICAL (F)
      DIMENSION HCH(20)
1     READ (*,2) (HCH(J), J=1,20)
2     FORMAT (20A1)
      IVAL=0
      FPOS=.TRUE.
      FSTRT=.FALSE.
C     INITIALIZE ZERO IN TOTAL, POSITIVE
C     SIGN, AND NO CHARACTERS SEEN YET

      DO 9 J=1,20
      IDG=IDV(HCH(J))
      IF (IDG) 10,11,11
10    WRITE (*,3)
3     FORMAT ('ERROR-REPEAT ENTRY '$)
C     CHECK FOR INVALID CHARACTERS
      GOTO 1
C     ACCEPT DIGITS 0...9, .-+ AND SPACE
C     NOTE: . SIGNALS END OF INPUT

11    IF (IDG-10) 12,13,14
C     13'.' MARKS END OF ENTRY - ANY
C     MORE CHARACTERS ARE IGNORED!

14    IF (IDG.EQ.13) GOTO 9
C     IGNORE SPACES ANYWHERE IN FIELD
      IF (FSTRT) GOTO 10
C     +- INVALID AFTER START

      IF (IDG.EQ.12) GOTO 16
      FPOS=.FALSE.
16    FSTRT=.TRUE.
      GOTO 9
C     +- SIGN CHECKED. STARTS NUMBER

12    TEMP=IDG+10.0*IVAL
C     SUM UP DIGITS TIMES POSITIONS
C     IMPLIED FLOAT
      IF (TEMP.GT.32767.0) GOTO 10
C     TOO LARGE FOR AN INTEGER
      IVAL=IFIX(TEMP)
      FSTRT=.TRUE.
9     CONTINUE

13    IF (FPOS) GOTO 20
      IVAL=-IVAL
C     CHECK SIGN OF TOTAL
20    INPUT=IVAL
      RETURN

      END

      FUNCTION VALIN()
C     FREE FORMAT FLOATING POINT NUMBER
C     ENTRY. DECIMAL POINT IS OPTIONAL.
C     RETURN YIELDS ZERO. SPACES IGNORED.
      IMPLICIT CHARACTER (H), LOGICAL (F)
      DIMENSION HCH(20)
```

CONTINUED ON PAGE 7

FROM PAGE 6


```
1     READ (*,2) (HCH(J), J=1,20)
2     FORMAT (20A1)
      FPOS=.TRUE.
      FSTRT=.FALSE.
      FDCPT=.FALSE.
      DIVI=10.0
      VLI=0.0
      VLD=0.0
C     INITIALIZE FLAGS FOR +/-, START OF
C     CHARACTERS IN FIELD, AND FINDING
C     DEC.PT. VLI=INTEGER PART, VLD=DEC.

      DO 9 J=1,20
      IDG=IDV(HCH(J))
      IF (IDG) 10,11,11
10    WRITE (*,3)
3     FORMAT ('ERROR-REPEAT ENTRY '$)
      GOTO 1
C     FUNCTION IDV RETURNS -1 IF CHARAC.
C     IS NOT 0...9 OR +.- OR SPACE
C     FUNCTION THEN ASKS FOR RE-ENTRY

11    IF (IDG-10) 12,13,14
14    IF (IDG.EQ.13) GOTO 9
C     IGNORE SPACES ANYWHERE IN FIELD

      IF (FSTRT) GOTO 10
C     +,- ONLY LEGAL AT START OF ENTRY
      IF (IDG.EQ.12) GOTO 16
      FPOS=.FALSE.
C     CASE OF NEGATIVE NUMBER
16    FSTRT=.TRUE.
C     IN ANY CASE, NUMBER HAS BEGUN
      GOTO 9

13    FDCPT=.TRUE.
      GOTO 9
C     DEP. PT. HAS BEEN ENCOUNTERED

12    FSTRT=.TRUE.
      IF (.NOT.FDCPT) GOTO 15
      VLD=VLD+IDG/DIVI
      DIVI=DIVI*10.0
      GOTO 9
C     ADD IN DECIMAL FRACTIONS FOR EACH
C     DIGIT PAST DECIMAL POINT

15    VLI=10*VLI+IDG
C     ADD IN DIGIT (FLOATED) AND SHIFT

9     CONTINUE
      VALIN=VLI+VLD
      IF (FPOS) GOTO 20
      VALIN=-VALIN
C     COMBINE INTEGER & DECIMAL PARTS
C     AND CHECK SIGN OF ENTRY
20    RETURN

      END


      FUNCTION IDV(HC)
C     RETURNS DIGIT VALUE 0...9, '.'=10
C     '+'=11, '-'=12, SPACE=13, ELSE=-1
      IMPLICIT CHARACTER (H)
      IDGT=ICHAR(HC)
      IF (IDGT.GE.48) GOTO 40
      GOTO (31,32,33,34) IDGT-42
      IF (IDGT.NE.32) GOTO 29
      IDV=13
      RETURN
C     32 IS ASCII FOR SPACE

40    IF (IDGT.LE.57) GOTO 41
29    IDV=-1
      RETURN
C     NOT A VALID DIGIT

41    IDV=IDGT-48
      RETURN
C     ASCII 48=0, 49=1,... , 57=9

31    IDV=12
      RETURN
C     ASCII 43='-'

32    GOTO 29
C     ASCII 44 INVALID

33    IDV=11
      RETURN
C     ASCII 45='+'

34    IDV=10
      RETURN
C     ASCII 46='.'

      END
```

ST. VALENTINES DAY MASSACRE DINNER

This is a reminder that the dinner for club members is scheduled for Saturday night February 14, 1981. Details were contained in last month's Harvest.

Marty Rutstein, who is organizing the affair, requests that all who are interested in attending send their checks for $25.00 per couple or $12.50 per single immediately. Information may be obtained by calling Marty at 845-2524.

QUICKIE HARDWARE REVIEW

I wanted to throughly review my newest purchase, an IDS 460G Paper Tiger, in this months issue, but time has become so short that I can only say these few lines.

However, the proof of how much I like the printer, despite its software problems, is the nice quality of the type you are looking at. This entire newsletter was printer using the 460 set at 12 characters per inch.

Dave Alpert

IAC INDEX

The IAC has sent some more APNOTES. Here is the up to date index of what is now in our library.

A. Hardware Modifications
- 2. User Firmware (2716)
- 4. Adding Color to Hires
- 6. Five Additional Characters from the Keyboard

B. Basics
- 2. Applesoft Hires Routines
- 3. Applesoft Array Eraser
- 4. Applesoft Hires Screen Function
- 5. Generating Tones in Applesoft
- 6. Literal Input Routine
- 8. Print Using Simulator
- 9. Converting Integer Basic Programs To Applesoft
- 10. Applesoft Random Numbers
- 11. Applesoft Out of Memory
- 12. VTAB and HOME Convertor for Sup-R-Terminal

C. Machine Language
- 8. Adding Features to LISA
- 9. Putting Programma M/L onto disk

D. Interfacing Information
- 2. Cassette Interface
- 4. DEL Character Killer
- 5. Correction to Sur-R-Terminal Preliminary Manual

E. Printer Interfacing
- 3. Serial Handshake Modification With Tabs

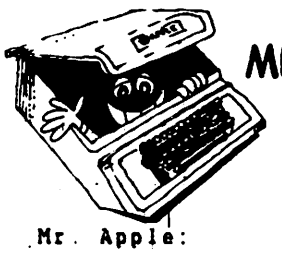
F. Datasheets
- 5. Apple Post

G. Pascal
- 1. Lower Case Patch
- 2. Linefeed
- 3. Take 280
- 4. Getrem
- 5. Transfer and Sum 512
- 7. Foreign and Gettext
- 9. Comm Card Baud Rate Changer
- 10. Interfacing Foreign Hardware
- 11. Long Integer Fix
- 12. Hires
- 13. Known Pascal Bugs
- 16. Pascal Units
- 17. Pascal Peeks and Pokes

H. Text and Graphics Information
- 1. Text screen Mapping and Use

I. Lists
- 1. DOS Demo Programs
- 2. 3.2.1 Append Fix

J. DOS



ASK MR. APPLE

Mr. Apple:

I have seen in many articles and programs were they have a statement like:

10 IF PEEK(2048) THEN GOTO 7 or

100 IF PR THEN PRINT "PR#1"

What I don't seem to understand is how you can have an IF statement without comparing it to something else. (eg. IF A = 7 THEN GOTO 250). What is even more confusing is that these programs run! Hope you can shed a little light on this matter.

Answer:

There really is no magic to these statements and they are really a shorthand way, and faster way of saying the following:

10 IF PEEK(2048) > 0 THEN GO TO 7

100 IF PR > 0 THEN PRINT "PR#1"

In essence what these commands allow the programmer to do is a fast compare of a value against 1 or 0. I use the word fast because the processor only has to look at one bit in its status register to make the decision whether to match on the value or reject it. In the later case, where a value is actually compared against, it might require many bytes to make the compare, depending on whether or not the value is an integer or a real number. Another example of this technique can be seen below:

10 IF NOT PR THEN 250

What this example means is if PR is not > 0, or in other words, if PR = 0 then the IF statement is satisfied and a jump will be made to line 250. I hope that these example clear up your questions.

Mr. Apple

CHANGING VOLUME NUMBERS

Tim Hartley
Hartley Software
3268 Coach Lane #2A
Kentwood, Mi. 49508
(616)942-8887

Often we want to change the volume number of a diskette so we might be able to differentiate it from one that currently has the same number.

Obviously one solution is to re-initialize all of our disks and use new volume numbers. Then we could transfer all programs from the old disk to the new. Alas, this is very tedious and time consuming. The solution, as well as the problem, lies in the fact that the volume number is written in the header for each and every sector on the diskette when it is first initialized. Then, anytime DOS accesses a sector, the volume number found in that sector's header is stored in one of DOS's little memory locations. I did devise a way of using a 'pseudo' volume number that works quite well.

First of all, here are two POKEs which should be either part of the HELLO program or POKEd in manually. Once they have been entered, the CATALOG command will display the "pseudo" number instead of the real volume number encountered for the sector.

```
POKE 44476,183
POKE 44477,179
```

Here is a short program which will allow you to modify the 'pseudo' number on all of your disks:

```
]LIST

0  TEXT : HOME
10  POKE 44476,183: POKE 44477,179
20  CALL 45047: REM READS IN VTOC
30  INPUT "ENTER NEW VOLUME NUMBER ";VL
40  POKE 46017,VL:
50  CALL 45051:
60  PRINT  CHR$ (4)"CATALOG"
70  END
```

These are for a 48K machine and CALLs must be modified for a smaller one. I haven't had a chance to try it on DOS 3.3 yet but I'm pretty sure it will work OK. If you fail to do the two POKEs, the old volume will appear. If you do the two POKEs, the 'new' number comes up.

Ed. note. I tried it on DOS 3.3 and it works fine. DMA

TIPS FROM GUY

Want to use commas (,) or colons (:) in strings with Applesoft INPUT statements? Simply preceed the input with a single quote mark character:

```
]LIST
10  INPUT A$
20  PRINT A$

]RUN
?THIS, FRIEND, IS: A TEST
?EXTRA IGNORED
THIS

]RUN
?"THIS, FRIEND, IS: A TEST
THIS, FRIEND, IS: A TEST
```

Note that the leading quote mark is ignored. A trailing quote mark is also ignored, if provided.

Now you want to save that string to a disk file and read it back later? If so, then you must preceed the string with a quote mark when writing the string to disk. This provides the quote mark needed by the INPUT statment when the string is later read back from disk. A CHR$(34) generates a quote mark in Applesoft.

```
.....
PRINT D$"WRITE ANY-OLD-FILE"
PRINT CHR$(34);A$
PRINT D$
.....
```

Remember, the quote mark generated by the CHR$(34) function will take a byte of diskette storage in the record!

Guy A. Lyle

BACKWARDS APPLE

Dr. Wm. R. Dial
438 Roslyn Ave.
Akron, Oh. 44320

Dr. Dial writes the 6502 Bibliography for MICRO magazine. This monthly feature lists articles that appear in the different periodicals that cover the 6502 microprocessor (which, of course, include the APPLE). The Harvest is now being included in that listing.

Dr. Dial was kind enouugh to send us this little program. I won't tell you what it does. Try it and see.

STATIC ELECTRICITY

Dave Alpert

I can't help but be redundant when I warn people of the dangers of static electricity around the APPLE. The APPLE is very sensitive to jolts of power generated on your person from just walking around the room or rubbing against your sweater, etc. It is very possible to put your APPLE out of commission with just one slight jolt of static. Touching the door of your disk drive when not properly discharged can blow away some of the chips on the disk controller card as well as reach into the motherboard itself.

The most important step to take to avoid disaster is to remember that the problems exists. It is to late to do anything after you hear and feel that zap go through your equipment. Grounding yourself before touching anything will usuallly prevent the problem. It is important to remember that, often, grounding yourself to something connected to the same electrical line as your equipment may result in the charge going through the line to the power supply and, hence, to the cards and chips on the board. Keep as far away from the APPLE as you can when discharging yourself.

Other preventative measures that can be taken are: spraying carpets with anti-static compounds, using anti-static floor mats and keeping the 'computer room' well humidified. Be careful not to over humidify the room as the APPLE is also sensitive to too much moisture in the air (to say nothing of what that does to wallpaper, furniture, etc.).

Probably the best solution is to temporarily relocate during the dry winter months in Florida or California. For those who can not do this, reread paragraphs 3 and 4 above.

```
 5  HOME : B = - 1: C = 1: PRINT "I
    NPUT STRING YOU WANT TO PRIN
    T": VTAB 6: INPUT A$: A = LEN
    (A$): HOME : VTAB 10: FOR I =
    A TO C STEP B: HTAB I: PRINT
     MID$ (A$,I);: GOSUB 40: NEXT
    I
 25  REM TRY A POKE 33,38 HERE AND
     SEE WHAT HAPPENS (THE EDITO
    R)
 30 B$ = "SOMETHING MUST BE WRONG
    WITH APPLE !": PRINT : PRINT
    : PRINT : E = LEN (B$): FOR
    I = E TO C STEP B: HTAB I: PRINT
     MID$ (B$,I);: GOSUB 40: NEXT
    I: PRINT : PRINT : PRINT : END

 40  FOR D = 1 TO 100: NEXT D: RETURN
```

CALL FOR PAPERS

The NCC or National Computer Conference will be held in Chicago next May 4-7. Along with the NCC will be the Personal Computing Festival (as well as the International APPLE Core meetings-see related article). The Festival will include three full days of programs and special activities on all aspects of personal computing, with emphasis on the applications people have for their own computers.

You can participate in one of four ways:

1. Pesent a paper.
2. Give a talk.
3. Organize a panel.
4. Deliver a tutorial.

Potential participants should send a letter of intent as soon as possible, but no later that December 1st. The letter should include an abstract and a brief biography.

Since you will receive this notice with only a week to go before the deadline date, I suggest a call be made to James Gerdes, Argonne National Laboratory, 9700 Cass Ave., Argonne, Il. 60439. 972-2000.
More information on this can be obtained by calling me at 295-6078.

Dave Alpert

NSAUG TO HOST IAC MEETING

Along with the NCC and The personal Computing Festival, the International APPLE Core will hold its annual meeting and seminars in Chicago the weekend of May 2 and 3 1981.

Our club has volunteered to be the 'host' club. We will coordinate finding facilities and lining up speakers for the seminars. A committee is being formed now to fulfill these needs. I am heading up the effort and could use help from those who are knowledgable in the areas of convention planning and seminar planning. We also would like to have several hard working volunteers who would like to learn how to do these things.

Please contact me immediately as time is short and any delay in making arrangements for this will be costly.

Dave Alpert

BIG CONTEST ENTERS STRETCH

How are those programs coming? The response of the stores throughout the area has been very favorable. It's now up to us to show the quality and quantity of our work. Besides the first prize of $100 and the second prize of $50, area stores have made the following contribution;

- Compushop-Rolling Meadows- coupons for yearly magazine subscription
- Compushop-Morton Groove-3 ,coupons for yearly magazine subscription-one copy MP Software graphics package-one Powersoft Super Checkbook
- Computerland-Arlington Heights-one $50 gift certificate-ten $10 gift certificates
- Computerland-Downers Groove-five $10 gift certificates
- Computerland-Niles-one 12" Leedex Video Monitor
- Data Domain-Schaumberg-$50 gift certificate
- Northbrook Computer-$25 gift certificate
- Video Etc.-Deerfield-$40 gift certificate

The club wishes to thank the participating dealers for their assistance and generosity.

Final deadline for the contest has been extended to include the January meeting to allow you to deliver your entries at that time. Remember, last day is January 3, 1981.

WANT ADS

Programmer wanted.
I am looking for a serious programmer willing to write APPLE II software for cash. Call Skip Neiburger 223-5077

Pete Zaloga is looking to buy a disc drive and 16K of memory. He can be reached at work at 545-8481 and at home at 283-6444.

NOT ANOTHER HELLO PROGRAM!

This hello program is for those who have an APPLE][PLUS and would like to automatically run their programs from a catalog by typing in a single letter. (An INTEGER version of a similar program was on NSAUG VOL 5 called NSAUG VOL 5, but this is no help to those who don't have INTEGER BASIC.)

To use this program, type it into your computer using FP (or APPLESOFT). Delete your old HELLO program, and save this program using the same name as your old HELLO program. You can now modify lines 10 to 30 to print any title you want.

BINARY files are loaded, and the program returns to the CATALOG for a new selection. Line 59240 may be changed if you want to BRUN binary files.

TEXT files are EXECed and the HELLO program removes itself from the result. The high line numbers are used so as not to interfere with line numbers on files being EXECed. Use the RENUMBER program on your master disk if you want to change them.

One word of warning: If the program you are trying to BLOAD resides at $800, it will blow the lines from 59280 on down and an error will occur. The binary program will be loaded however. This is because APPLESOFT programs begin there and will be wiped out by the BLOAD.

```
]LIST

59000 T$ = "AUTO-HELLO"
59010 N$ = "A. F. PETRIE"
59020 D$ = "9/27/80"
59030  TEXT : HOME
59040  HTAB 3: FOR I = 3 TO 37: PRINT
     "*";: NEXT
59050  PRINT : PRINT : PRINT  TAB(
     20 -  INT (.5 *  LEN (T$)));
     T$
59060  PRINT : PRINT  TAB( 20 -  INT
     (.5 *  LEN (N$)));N$
59070  PRINT : PRINT  TAB( 20 -  INT
     (.5 *  LEN (D$)));D$
59080  PRINT : HTAB 3: FOR I = 3 TO
     37: PRINT "*";: NEXT
59090  PRINT
59100  FOR I = 1 TO 1000: NEXT I
59110 D$ =  CHR$ (4)
59120  DEF  FN S(H) =  SCRN( H,2 *
     V) + 16 *  SCRN( H,2 * V + 1
     )
59130  REM FNS(H)=ASC() FOR CHAR
     AT H,V
59140  PRINT D$;"CATALOG"
59150 LAST =  PEEK (37): REM CURS
     ER V POS
59160 V = 0
59170 V = V + 1: IF  FN S(1) = 16
     0 GOTO 59170: REM LOOK FOR F
     IRST LINE IN CATALOG
59180  IF  FN S(2) <  > 160 THEN
     V = V + 2: REM ADD 2 IF ONLY
      TITLE
59190  FOR I = V + 1 TO LAST: VTAB
     I: HTAB 3: PRINT "- "; CHR$
     (64 + I - V);" -": NEXT I: REM
     PRINT LETTERS
59200  PRINT "CHOOSE LETTER (Z=EX
     IT): ";: GET N$
59210  IF  ASC (N$) < 65 THEN  END

59220 N =  ASC (N$) - 64: IF N >
     LAST - V THEN  END
59230 V = V + N - 1:A$ = ""
59240 N$ = "RUN ": IF  FN S(1) =
     194 THEN N$ = "BLOAD": GOTO
     59270
59250  REM CHANGE "BLOAD" TO "BRU
     N" IF DESIRED
59260  IF  FN S(1) = 212 THEN N$ =
     "EXEC "
59270  FOR H = 7 TO 39:A$ = A$ +
      CHR$ ( FN S(H)): NEXT
59275  PRINT
59280  PRINT D$;N$;A$
59290  IF  LEFT$ (N$,2) = "BL" THEN
      HOME : PRINT  TAB( 3)")) BI
     NARY FILE LOADED <<": GOTO 5
     9140
59300  IF  LEFT$ (N$,1) = "E" THEN
      DEL 59000,59300: REM  REMOV
     E THIS PROGRAM FROM TEXT FIL
     E
```

RAP RAP YOUR APPLE

Joe Rakosky has come up with a novel way to protect your keyboard from being soaked when you spill your favorite cola n it. He covered the keyboard with Saran Wrap. Joe tells us he has no trouble typing on the keyboard and the wrap keeps the electrical contacts dry.

HARDWARE REVIEW
California Computer Systems
7424 Calendar/Clock Module

Dav Holle
Illini MicroComputers, Naperville, Illinois

When Mountain Hardware raised the price of their Apple Clock to $280, I had given up on ever having a clock in my Apple. Why so expensive? With low-cost digital watches all over the place, there had to be a cheaper way. About a month ago, California Computer Systems proved it: they introduced their 7424 Calendar/Clock Module for $125. Even at less than half the price of the Mountain Hardware clock, it's a full feature unit, and could have been even cheaper if some features were left out.

I've used the Mountain Hardware Apple Clock for over a year in the Naperville Apple Bulletin Board System, and now I have the CCS 7424 in my own machine. When I refer to the 'Apple Clock', that's the Mountain Hardware board!

The Apple Clock has a rechargeable battery that lasts 'up to four days' (usually less). The CCS clock won't go on vacation whenever you do, though. To keep it running while the Apple's off, all that's required is 2 watch cells that keep it running for a year.

The Apple Clock keeps a 24-bit count of seconds and a 3-digit milliseconds count, which must be converted to month/day hour:min:sec by the firmware each time it is read. Instead of doing that, the CCS clock hardware actually keeps track of the time, day of week and date by itself. If you wish, it can be read without using the card's firmware at all. Reading from BASIC is simple: a poke to select which digit, and a peek to read it. The 7424 keeps track of the year and the day of the week if you set them. The Mountain Hardware clock has no way to do this directly.

There are several sets of solderless jumpers on the 7424. These control write-protecting the clock setting (so turning the Apple off doesn't reset it), interrupt interval, ROM or RAM program, and which program in the 2708 EPROM is in control. Most jumpers, the write-protect and program-select in particular, can be moved without taking the card out.

If you wish to use the clock to generate interrupts, you may select 1/1024 sec., 1 sec., 1 minute or 1 hour intervals. If you choose, you may take the out the EPROM provided and put 256 bytes of RAM in the sockets provided. You'll then have to move 1 jumper to use RAM instead of ROM, for those who just have to do things their own way!

Normally, you would select one of the 3 EPROM programs already on the 2708. (Brave souls are free to burn their own into the blank 4th area.) Most people will probably use the 'clock input' routine. This program produces a string similar to the Apple Clock's when you select the clock for input. The 7424 can give you the year & the day of the week, which aren't available from the Apple Clock, so the 'clock input' routine wastes some abilities of the card for compatibility's sake. The milliseconds part of this time string will always be 000, since fractions of a second aren't available on the 7424.

The clock input routine is not totally compatible with the Apple Clock. When using an Apple Clock in slot 4, the usual practice is to set both input and output to the clock like this:

```
1000 IN#4:PR#4:INPUT " ";TI$:
     REM Apple Clock method
```

The CCS clock firmware does NOT allow you to set output to it, and if you do, it goes nuts. So, the 7424 is almost a direct replacement for the Apple Clock, but not quite. Take out the PR# to the clock and it'll work:

```
1000 IN#4:INPUT TI$:
     REM CCS 7424 Method
```

I don't know if this makes it incompatible with Mountain Hardware-BSR control system or not. If the home control system does simple reads of the clock, there should be little difficulty in converting it.

The other programs you may select make use of the interrupts. The 'time string' routine, for Applesoft only, can continuously update a string (TI$) in memory with the current time. The 'time display' routine will show the time on the screen. The rate of the updates will depend on the interrupt interval selected by your jumpers.

The EPROM routines provided never make use of the $C800-$CFFF shared ROM area, so there's no problem when using clock interrupts with other cards that use that area. This lets you use cards like the Apple Serial Card, the DC Hayes Micromodem II, the Mountain Hardware ROM+ or the Silentype while interrupts are enabled. By contrast, the Mountain Hardware Apple Clock deselects other cards' shared ROMs. This means that none of the above cards can be in use while interrupts are enabled with the Apple Clock, but are OK with the 7424.

FROM PAGE 0D

The manual is usable but weak in some areas: it does not tell you how to set the day of the week. The clock setting routine given doesn't test to make sure eaoh digit is set properly, which lead to some difficulty when I found that one of the the year digits on my clock was stubborn and needed more than one poke to set it. The CCS people were very helpful though, and quickly solved the problem over the phone.

All in all, I believe the 7424 Calendar/Clock Module is the best bet for most applications. Where fractions of a second or a larger selection of interrupt intervals are required, one might be better off with the more expensive Apple Clock. But, in the usual case where the the date and time is needed for reports or files, the CCS version makes a lot of sense.

EDUCATIONAL SOFTWARE SYMPOSIUM

Monica Kantrowitz, President of Queue informs us that a Educational Softwsre Symposium will be held at the O'Hare Hilton on February 7 and 8, 1981. Seminars will be conducted and will include the following topics;

Educational Software for Elementary Schools
Educational Software in the Mathematic Cirriculum
Educational Software in the Science Currirrulum
Computer Assisted Instruction in Foreign Languages
Can Computers Teach English and Reading?Simulations
Computer Education
Writing Educational Software
New Technological Developments

There will also be an extensive exhibit hall and an opportunity to view software programs first hand. Registration is $85. Write Monica at 5 Chapel Hill drive, Fairfield, Ct. 06432

STOP (AND START) LIST

A. F. Petrie

Ever watch helplessly while important information in your print-out or program listing scrolled off the screen? The solution is simple.

If you have an INTEGER APPLE][(no plus) there are two programs on NSAUG VOL 3 Disk which you can BRUN at any time without losing your current program. To use them, simply pull out the disk that you are currently using, insert NSAUG VOL 3, and BRUN STOPLIST or BRUN SLOW LIST and you can halt print-outs or listings by hitting any key. You can resume by hitting another key. STOPLIST clicks every letter, SLOW LIST is the silent partner.

If you have an APPLE][PLUS, life is even easier. All you have to do is hit CTRL-S to stop and any other key to continue. If you want to stop the program or listing (for example to modify it) hit CTRL-C after CTRL-S. If there are no changes, you can resume by typing CONT in APPLESOFT or CON in INTEGER. I usually hold down CTRL with the third finger of my left hand and keep the second finger over S before pushing RETURN, so I can stop instantly. This piece of information is hidden on pages 28 and 30 of the NEW APPLE][REFERENCE MANUAL. There is only one way to stop a CATALOG listing that is too long for the page: use RESET and the list will stay. Sorry for you non-PLUS owners, this won't work for you!

Merry Christmas to all

the Editor

Copy 1

NSAUG DISK ORDER FORM

Please fill out and give (or mail) top copy to the club librarian with the proper fee. If mail order, enclose check only. NO CASH. Indicate the volume numbers of the disk you want below.

( ) ( )

Date ordered_____

Total no. of disks ordered_____

* $3.50=$_____

Mail svc.* $1.50/ =$_____

Total enclosed $_____

* Librarian's use only *

Date (ready-mailed)_____/_____/_____

( )Cash ( ) Amount $_____

Name(please PRINT)_____Address_____

_____City_____State_____Zip_____

********** TEAR HERE **********

Copy 2

This copy for your records.

Disks ordered:

( ) ( )

Date submitted_____Amount paid_____

FIRST CLASS MAIL

MICHAEL L ROBINS
1300 S ELMHURST RD
MT PROSPECT, IL 60056
81

THE HARVEST NEWSLETTER
N.S.A.U.G
880 MELLODY
LAKE FOREST, IL 60045S